MODELING, MANAGING, AND SCHEDULING VIDEOCONFERENCE MEETINGS

Inventors: Chia-I-Hsien Li, San Jose, CA (US); Brian Chan, Palo Alto, CA (US)

Correspondence Address:
EPSON RESEARCH AND DEVELOPMENT INC
INTELLECTUAL PROPERTY DEPT
2580 ORCHARD PARKWAY, SUITE 225
SAN JOSE, CA 95131 (US)

Publication Classification

Abstract

Embodiments of the present invention provide data structures for modeling videoconference meetings, including videoconference meetings having only a single occurrence and videoconference meetings that repeat regularly, such as on a weekly or monthly basis. Some embodiments additionally model videoconference meetings that are part of a repeating series, but that have been canceled or moved to a different time, day, or the like, than the other videoconference meetings in the series. Embodiments of the present invention also provide methods for managing videoconference meetings, scheduling videoconference meetings, displaying the schedules, and monitoring videoconference participant load for videoconference meetings. Some embodiments additionally model videoconference media files, and methods for managing the media files.
FIG. 1
FIG. 2
300

Select coverage period comprising plurality of intervals.

302

Select first interval and first videoconference meeting entity.

304

Repeating?

306

Yes

Branched removed?

Yes

308

No

314

Done

316

Last interval?

Yes

310

Last meeting?

Yes

312

Branched?

Yes

318

Intersect?

Yes

320

Display videoconference meeting.

322

FIG. 3
500

Store media file.

502

Generate media entity for videoconference meeting entity.

504

Branch?

506

Yes

Generate media entity for branch entity.

508

No

506

Done

510

FIG. 5
Receive request to delete media entity.

Interval type? [Decide whether the interval is once only or repeating.]

- If once only or repeating: Continue to the next step.
- If branched: Retain media file.

Delete media file.

FIG. 6
Select videoconference meetings that intersect interval.

Generate array of data pairs.

Sort array of data pairs.

Select first data pair in array.

Add participant value to participant total.

Total > max? Yes

Indicate videoconference load violation.

No

Next data pair.

Last data pair?

No

Yes

Done

FIG. 7
FIG. 8
MODELING, MANAGING, AND SCHEDULING VIDEOCONFERENCE MEETINGS

BACKGROUND

[0001] The present invention relates generally to videoconferencing. More particularly, the present invention relates to modeling, managing, and scheduling videoconference meetings.

SUMMARY

[0002] In general, in one aspect, the invention features a data structure comprising a plurality of videoconference meeting entities each representing a videoconference meeting; and a plurality of attributes for each of the videoconference meeting entities, the attributes comprising one or more schedule attributes each having a value selected from the group consisting of a scheduled start time of the respective videoconference meeting, a scheduled end time of the respective videoconference meeting, and a scheduled duration of the respective videoconference meeting, and an interval type attribute having a value selected from the group consisting of once only, indicating that the respective videoconference meeting comprises only one occurrence, repeating, indicating that the respective videoconference meeting comprises a repeating series of occurrences, and branched, indicating that the respective videoconference meeting comprises only one occurrence, and that the attributes of the respective videoconference meeting entity have the same values as the attributes of another one of the videoconference meeting entities having an interval type attribute value of repeating except for one or more of the schedule attributes.

[0003] In some embodiments, each of the videoconference meeting entities has a different meeting identifier, and each occurrence of the videoconference meetings has a different index. In such embodiments, the data structure further comprises a branch entity representing a relationship between a first one of the videoconference meeting entities having an interval type attribute value of repeating and a second one of the videoconference meeting entities having an interval type attribute value of branched; and a plurality of attributes for the branch entity, the attributes comprising a series identifier having a value representing a meeting identifier of the first one of the videoconference meeting entities, a branch identifier having a value representing a meeting identifier of the second one of the videoconference meeting entities, and an index of a videoconference meeting represented by the second one of the videoconference meeting entities.

[0004] In general, in one aspect, the invention features a method for displaying a schedule of the videoconference meetings, the method comprising: selecting a coverage period for the schedule of meetings, the coverage period comprising a plurality of intervals; and for each of the intervals in the coverage period, and each of the videoconference meeting entities having an interval type attribute value of repeating, displaying an indicator of the occurrence of the respective videoconference meeting when the occurrence of the respective videoconference meeting intersects temporally with the interval, and none of the series identifiers of the branch entities has a value representing the meeting identifier of the videoconference meeting entity representing the respective videoconference meeting. Some embodiments comprise determining whether the occurrence of the respective videoconference meeting intersects temporally with the interval, comprising at least one step selected from the group comprising determining whether the scheduled start time of the respective videoconference meeting follows an end time of the interval, and determining whether the scheduled end time of the respective videoconference meeting precedes a start time of the interval. Some embodiments comprise a medium or waveform containing a program of instructions, e.g., a computer program, for performing the method. Some embodiments comprise an apparatus to perform the method.

[0005] Some embodiments comprise a plurality of media entities each representing a link between a media file and one of the videoconference meeting entities. Some embodiments comprise a method for managing the media entities, comprising: storing one of the media files; generating a first one of the media entities representing a link between the one of the media files and a selected one of the videoconference meeting entities; and generating one or more second ones of the media entities each representing a link between the one of the media files and one of the branch entities having series identifier values representing the meeting identifier of the selected one of the videoconference meeting entities. Some embodiments comprise deleting the one of the media files when the first one of the media entities is deleted. Some embodiments comprise retaining the one of the media files when any of the second ones of the media entities are deleted. Some embodiments comprise a medium or waveform containing a program of instructions, e.g., a computer program, for performing the method. Some embodiments comprise an apparatus to perform the method.

[0006] Some embodiments comprise a plurality of invited videoconference participant entities each representing a videoconference participant invited to one of the videoconference meetings. Some embodiments comprise a method for calculating a videoconference participant load for an interval in a videoconference schedule comprising a plurality of the videoconference meetings, the method comprising: generating an array comprising, for each of the videoconference meetings having an occurrence that temporally intersects the interval, generating a first data pair comprising a first participant value representing the number of the participant entities associated with the videoconference meeting and a first difference between a reference time and the start time of the videoconference meeting, and generating a second data pair comprising a second participant value representing the negative of the number of the participant entities associated with the videoconference meeting and a second difference between the reference time and the end time of the videoconference meeting; and traversing the array in order according to the first and second differences, comprising adding the first and second participant values to a participant total. Some embodiments comprise indicating a videoconference load violation when the videoconference participant total exceeds a predetermined maximum participant total.

[0007] In general, in one aspect, the invention features a method for calculating a videoconference participant load for an interval in a videoconference schedule comprising a plurality of the videoconference meetings, the method comprising: generating an array comprising, for each of the videoconference meetings having an occurrence that tem-
porally intersects the interval, generating a first data pair comprising a participant value representing the number of videoconference participants associated with the videoconference meeting and a first difference between a reference time and the start time of the videoconference meeting, and generating a second data pair comprising a participant value representing the negative of the number of the videoconference participants associated with the videoconference meeting and a second difference between the reference time and the end time of the videoconference meeting; and traversing the array in order according to the first and second differences, comprising adding the first and second participant values to a participant total.

[0008] Some embodiments comprise indicating a videoconference load violation when the videoconference participant total exceeds a predetermined maximum participant total. Some embodiments comprise a medium or waveform containing a program of instructions, e.g., a computer program, for performing the method. Some embodiments comprise an apparatus to perform the method.

[0009] The details of one or more implementations are set forth in the accompanying drawings and the description below. Other features will be apparent from the description and drawings, and from the claims.

DESCRIPTION OF DRAWINGS

[0010] FIG. 1 shows a videoconference system comprising a videoconference server and a plurality of videoconference clients. Also shown is an optional media server.

[0011] FIG. 2 shows a data structure for modeling videoconferences in the videoconference system according to a preferred embodiment of the present invention.

[0012] FIG. 3 shows a process for the videoconference system for displaying a schedule of videoconference meetings according to a preferred embodiment of the present invention.

[0013] FIG. 4 shows an exemplary videoconference meeting schedule.

[0014] FIG. 5 shows a process for the videoconference system for adding media files for videoconference meetings according to a preferred embodiment of the present invention.

[0015] FIG. 6 shows a process for the videoconference system for deleting media files for videoconference meetings according to a preferred embodiment of the present invention.

[0016] FIG. 7 shows a process for the videoconference system for calculating a videoconference participant load for an interval in a videoconference schedule comprising a plurality of the videoconference meetings according to a preferred embodiment of the present invention.

[0017] FIG. 8 shows an example videoconference schedule comprising a plurality of videoconference meetings that is used to illustrate the operation of the process.

[0018] The leading digit(s) of each reference numeral used in this specification indicates the number of the drawing in which the reference numeral first appears.

DETAILED DESCRIPTION

[0019] Embodiments of the present invention provide data structures for modeling videoconference meetings, including videoconference meetings having only a single occurrence and videoconference meetings that repeat regularly, such as on a weekly or monthly basis. Some embodiments additionally model videoconference meetings that are part of a repeating series, but that have been canceled or moved to a different time, day, or the like, than the other videoconference meetings in the series. For example, a repeating Monday videoconference meeting may occasionally be moved to the following Tuesday to accommodate a Monday holiday. Embodiments of the present invention also provide methods for managing videoconference meetings, scheduling videoconference meetings, displaying the schedules, and monitoring videoconference participant load for videoconference meetings. Some embodiments additionally model videoconference media files, for example representing charts, video clips, and the like, to be shared with videoconference meeting participants. Embodiments of the present invention also provide methods for managing the media files.

[0020] FIG. 1 shows a videoconference system 100 comprising a videoconference server 102 and a plurality of videoconference clients 104A-N. Also shown is an optional media server 106. Preferably videoconference server 102, videoconference clients 104, and optional media server 106 are connected by one or more networks 108 such as the Internet.

[0021] FIG. 2 shows a data structure 200 for modeling videoconferences in videoconference system 100 according to a preferred embodiment of the present invention. Data structure 200 comprises one or more videoconference meeting entities 202 each representing a videoconference meeting, and a plurality of attributes 204 for each of the videoconference meeting entities 202. Attributes 204 comprise an optional meeting identifier (Meet-ID) 204A that identifies the videoconference meeting represented by the respective videoconference meeting entity 202, an optional name attribute (Name) 204B that identifies a name of the respective videoconference meeting entity 202, an optional owner identifier attribute (Owner-ID) 204C that identifies an owner of the videoconference meeting represented by the respective videoconference meeting entity 202, an optional security attribute (Security) 204D that identifies security attributes of the videoconference meeting represented by the respective videoconference meeting entity 202, one or more schedule attributes 204E that identify a schedule of the videoconference meeting represented by the respective videoconference meeting entity 202, and an interval type attribute 204F that identifies any repetitions of the videoconference meeting represented by the respective videoconference meeting entity 202.

[0022] Schedule attributes 204E preferably comprise a scheduled start time (Sched. Start Time) 204EA that identifies a scheduled start time of the videoconference meeting represented by the respective videoconference meeting entity 202, a scheduled end time (Sched. End Time) 204EB that identifies a scheduled end time of the videoconference meeting represented by the respective videoconference meeting entity 202, and an optional scheduled duration (Sched. Duration) 204EC that identifies a scheduled duration of the videoconference meeting represented by the respective videoconference meeting entity 202.
Interval type attribute 204F preferably has a value selected from the group consisting of “once only,” indicating that videoconference meeting represented by the respective videoconference meeting entity 202 comprises only one occurrence, “repeating,” indicating that videoconference meeting represented by the respective videoconference meeting entity 202 comprises a repeating series of occurrences, and “branched,” indicating that videoconference meeting represented by the respective videoconference meeting entity 202 comprises only one occurrence, and that the attributes 204 of the respective videoconference meeting entity 202 have the same values as the attributes of another one of the videoconference meeting entities 202 having an interval type attribute value of “repeating” except for one or more of schedule attributes 204E. In some embodiments, the value of interval type attribute 204F comprises the duration of the repetition interval when it has a value of “repeating.” In other embodiments, the value of “repeating” is replaced with values representing the repetition interval, such as “daily,” “weekly,” and the like.

Occasionally it is desirable to reschedule an occurrence of a repeating videoconference meeting. Data structure 200 optionally further comprises branch entities 206 to represent the rescheduled occurrence, which is referred to herein as a “branched” videoconference meeting, and the relationship between the branched videoconference meeting and the associated repeating videoconference meeting. Referring again to FIG. 2, each branch entity 206 represents a relationship between a repeating videoconference meeting entity 202 (that is, a videoconference meeting entity 202 having an interval type attribute value of “repeating”) and a branched videoconference meeting entity 202 (that is, a videoconference meeting entity 202 having an interval type attribute value of “branched”). Data structure 200 optionally further comprises a plurality of attributes 208 for each branch entity 206. Attributes 208 comprise a series identifier (Series Meet_ID) 208A having a value representing a meeting identifier of the repeating videoconference meeting entity 202, a branch identifier (Branched Meet_ID) 208B having a value representing a meeting identifier of the branched videoconference meeting entity 202, and an index (Index) 208C of a videoconference meeting represented by the second videoconference meeting entity.

Each occurrence of the videoconference meetings represented by a videoconference entity 202 has a different index representing the order of occurrence. Tables 1 and 2 illustrate the use of the index according to a preferred embodiment of the present invention.

| Table 1 |
|---|---|---|---|---|
| Meet_ID | Name | Owner_ID | Sched. Start Time | Sched. End Time | Security | Interval Type |
| 9 | 'series' | 10 | 01/28/05 | 01/28/05 | true | weekly |
| | | | (Fri.) | (Fri.) | | |
| | | | 1:00 pm | 2:00 pm | | |
| 11 | 'branch' | 10 | 02/10/05 | 02/10/05 | true | branched_add |
| | | | (Thurs.) | (Thurs.) | | |
| | | | 3:00 pm | 4:00 pm | | |

This example represents a repeating meeting having one branch. Table 1 shows the attribute values of the videoconference meeting entities 202 for the repeating videoconference meeting and the branched videoconference meeting. Referring to Table 1, the repeating meeting has a meeting identifier of 9, and the branched meeting has a meeting identifier of 11. The repeating videoconference meeting occurs every Friday from 1:00 p.m.-2:00 p.m., beginning on Jan. 28, 2005. The branched videoconference meeting attributes show that the third occurrence of the repeating videoconference meeting has been moved to the preceding Thursday.

Table 2 shows the attribute values for the branch entity 206 representing the relationship between the repeating videoconference meeting and the branched videoconference meeting. Referring to Table 2, the branch entity 208 has a series meeting identifier of 9 that identifies the repeating meeting in Table 1, a branched meeting identifier of 11 that represents the branched meeting in Table 1, and an index of two, which shows that the branched meeting is the third occurrence (because the first index value is zero) of the repeating meeting.

Occasionally it is desirable to cancel an occurrence of a repeating videoconference meeting. Preferably a branched videoconference meeting entity 202 having an interval type attribute value of “branched_removed” is used to represent canceled occurrences of a repeating videoconference meeting.

Occasionally videoconference participants find it is desirable to share media files such as charts, video clips, and the like. Data structure 200 optionally further comprises media entities 210 each representing a link between a media file and a videoconference meeting entity 202. Referring again to FIG. 1, the media files are preferably stored on optional media server 106.

Data structure 200 also optionally includes participant entities 212 each representing a participant of one or...
more of the videoconference meetings represented by videoconference meeting entities 202.

[0032] Embodiments of the present invention provide a method for displaying a schedule of videoconference meetings for a coverage period comprising one or more intervals. In general, for each of the intervals in the coverage period, and each of the videoconference meeting entities 202 having an interval type attribute value of repeating, the method displays an indicator of the occurrence of the respective videoconference meeting when the occurrence intersects temporally with the interval, and is not a branch of the repeating videoconference meeting (that is, none of the series identifiers of the branch entities 206 has a value representing the meeting identifier of the videoconference meeting entity 202 representing the respective videoconference meeting). The pseudo-code that follows presents one implementation of the method.

```java
for every interval
  if (meeting repeats)
    calculate the closest index of the repeating series to the day
    if (a branch for the index exists)
      do not show the meeting;
      return;
  else
    if (meeting intersect with the interval)
      show the meeting;
      return;
  }
  else
    if (meeting's interval type is not BRANCH_REMOVE)
      if (meeting intersect with the interval)
        show the meeting;
        return;
    }
  }
  do not show the meeting;
  return;
```

[0033] FIG. 3 shows a process 300 for videoconference system 100 for displaying a schedule of videoconference meetings according to a preferred embodiment of the present invention. Process 300 selects a coverage period for the schedule of meetings, the coverage period comprising a plurality of intervals (step 302). For example, the coverage period can be one day, and each interval can be one hour.

[0034] Process 300 then selects the first interval of the coverage period and a first one of the videoconference meeting entities 202 (step 304). If the selected videoconference meeting entity 202 does not have an interval attribute value of “repeating,” process 300 calculates the index of the occurrence of the repeating meeting represented by the selected videoconference meeting entity 202 that is closest in time to the selected interval (step 324). The pseudo-code that follows presents one implementation of a method for calculating the index given the meeting identifier mt and the current time.

```java
int getSeriesIndex(Meeting mt, long tmTime) {
  int seriesIndex = 0;
  // Fix the time to the end of the day
  long tmEndOfDay = tmTime + 3600000 - tmTime % 3600000;
  // Calculate the interval
  switch (mt.getIntervalType( )) {
    case ONCE_ONLY:
    case BRANCH_ADD:
    case BRANCH_REMOVE:
      interval = -1;
      break;
    case DAILY:
      interval = DAY_INTERVAL;
      break;
    case WEEKLY:
      interval = 7 * DAY_INTERVAL;
      break;
    }
  // Calculate the index
  if (interval < 0) {
    seriesIndex = 0;
  } else {
    long distance = tmEndOfDay - mt.getScheduleTime();
    seriesIndex = (int) (distance / interval);
  }
  return seriesIndex;
```

[0035] But if at step 308 the selected videoconference meeting entity 202 does not have an interval attribute type value of “branched-removed” (indicating the value is either “once only” or “branched”) and intersects temporally (that is, overlaps in time) with the selected interval (step 320), then process 300 displays the videoconference meeting (step 322), for example on a meeting schedule such as that shown in FIG. 4. Otherwise process 300 resumes at step 310.

[0036] In general, to determine whether an occurrence of a videoconference meeting intersects temporally with the interval, process 300 determines whether the scheduled start time of the videoconference meeting follows the end time of the interval, and determines whether the scheduled end time of the videoconference meeting precedes the start time of the interval. If either condition is true, there is no intersection. The pseudo-code that follows presents one implementation of the method.

```java
if (tmIntervalStart < tmMeetingsStart + 1 millisecond)
  tmSSST = the meeting's scheduled start time
  tmSET = the meeting's scheduled end time
  if (tmIntervalStart < tmSSST || tmIntervalStart > tmSET) return false;

return true;
```
If the videoconference meeting entity 202 identified by the index obtained at step 324 has an interval type attribute value of "branched" (step 326), then process 300 continues at step 310. Otherwise, process 300 continues at step 320.

Embodiments of the present invention provide a method for managing media files for videoconference meetings according to a preferred embodiment of the present invention. FIG. 5 shows a process 500 for videoconference system 100 for adding media files for videoconference meetings according to a preferred embodiment of the present invention. In general, the media files are provided by one of the videoconference meeting participants, but may also originate elsewhere. Process 500 stores the media files (step 502), preferably on optional media server 106. Process 500 generates one or more media entities 210 each representing a link between one of the media files and one of videoconference meeting entities 202 (step 504). If a videoconference meeting entity represents a repeating meeting having one or more branches (step 506), then process 500 generates one or more media entities 210 each representing a link between the media file and one of the branch entities 206 of the repeating meeting (step 508) and is done (step 510). Otherwise process 500 is done (step 510). Branch entities 206 related to the repeating meeting can be identified by having series identifier values representing the meeting identifier of the videoconference meeting entity 202 representing the videoconference meeting.

FIG. 6 shows a process 600 for videoconference system 100 for deleting media files for videoconference meetings according to a preferred embodiment of the present invention. Process 600 receives a request to delete a media entity 210 (step 602), for example because the corresponding videoconference meeting entity 202 has been deleted. The corresponding media file can be deleted or retained according to one or more media file retention rules. For example, when the corresponding videoconference meeting entity 202 represents a once only or repeating meeting (step 604), the media file is deleted (step 606) because all occurrences of the corresponding videoconference meeting have been deleted. But if at step 604 the corresponding videoconference meeting entity 202 represents a branched meeting, the media file is retained (step 608) because the corresponding repeating meeting has not been deleted, and so has one or more remaining occurrences that could require the media file.

FIG. 7 shows a process 700 for videoconference system 100 for calculating a videoconference participant load for an interval in a videoconference schedule comprising a plurality of the videoconference meetings according to a preferred embodiment of the present invention. Some embodiments of process 700, such as the embodiment described below, employ data structure 200 of FIG. 2, while others do not, as will be apparent to one skilled in the relevant arts after reading this description.

FIG. 8 shows an example videoconference schedule 800 comprising a plurality of videoconference meetings A, B, C, D, E, and M that is used to illustrate the operation of process 700 of FIG. 7. The participant load calculation interval is depicted as two vertical lines, one at the start time SM, and the other at the end time, EM, of the interval. Note that the interval coincides with videoconference meeting M.
boolean isServerLoadExceeded(int maxConnections, long startTime, long endTime, int initialCount) {
    Vector vMeetings = query for all meetings;
    Vector vLoadCounts = new Vector(1);
    for (each meeting m in vMeetings) {
        int numParts = number of participants in meeting m;
        if (m intersects with startTime and endTime) {
            vLoadCounts.addElement(new LoadCount(meeting m's startTime, numParts));
        }
        vLoadCounts.sort(); // sort vLoadCounts according to the time field;
        int partsCount = initialCount;
        for (each loadCount in vLoadCounts) {
            partsCount += loadCount.partCount;
            if (partsCount > maxConnections) {
                return true;
            }
        }
        return false;
    }
}

where a LoadCount stores a pair of attributes long time and int partCount

Of course, some of the videoconference meetings may be repeating meetings, as described above. For each repeating meeting, process 700 determines whether any of the occurrences of the repeating meeting intersects temporarily with the interval, and selects those that intersect for inclusion in the load calculation. The pseudo-code that follows presents one implementation of the method.

[0057] The invention can be implemented in digital electronic circuitry, or in computer hardware, firmware, software, or in combinations of them. An apparatus of the invention can be implemented in a computer program product embodied in a machine-readable storage medium or device for execution by a programmable processor; and method steps of the invention can be performed by a programmable processor executing a program of instructions to perform functions of the invention by operating on input data and generating output. The invention can be implemented advantageously in one or more computer programs that are executable on a programmable system including at least one programmable processor coupled to receive and instructions from, and to transmit data and instructions to, a data storage system, at least one input device, and at least one output device. Each computer program can be implemented in a high-level procedural or object-oriented programming language, or in assembly or machine language if desired; and in any case, the language can be a compiled or interpreted language. Suitable processors include, by way of example, both general and special purpose microprocessors. Generally, a processor will receive instructions and data from a read-only memory and/or a random access memory. Generally, a computer will include one or more mass storage mediums or devices for storing data files; such devices include magnetic disks, such as internal hard disks and removable disks; magneto-optical disks; and optical disks. Storage devices suitable for embodying computer program instructions and data include all forms of non-volatile memory, including by way of example semiconductor memory devices, such as EPROM, EEPROM, and flash memory devices; magnetic disks such as internal hard disks and removable disks; magneto-optical disks; and CD-ROM disks. A waveform, e.g., a carrier wave may also serve as a storage medium for containing program instructions. Program instructions may also be supplemented by, or incorporated in, ASICs (application-specific integrated circuits).

[0058] A number of implementations of the invention have been described. Nevertheless, it will be understood that various modifications may be made without departing from the spirit and scope of the invention. Accordingly, other implementations are within the scope of the following claims.
a plurality of attributes for each of the videoconference meeting entities, the attributes comprising
one or more schedule attributes each having a value selected from the group consisting of
a scheduled start time of the respective videoconference meeting,
a scheduled end time of the respective videoconference meeting,
a scheduled duration of the respective videoconference meeting, and an interval type attribute having a value selected from the group consisting of
once only, indicating that the respective videoconference meeting comprises only one occurrence,
repeating, indicating that the respective videoconference meeting comprises a repeating series of occurrences, and
branched, indicating that the respective videoconference meeting comprises only one occurrence, and that the attributes of the respective videoconference meeting entity have the same values as the attributes of another one of the videoconference meeting entities having an interval type attribute value of repeating except for one or more of the schedule attributes.

2. The data structure of claim 1, wherein each of the videoconference meeting entities has a different meeting identifier, and wherein each occurrence of the videoconference meeting has a different index, further comprising:

a branch entity representing a relationship between a first one of the videoconference meeting entities having an interval type attribute value of repeating and a second one of the videoconference meeting entities having an interval type attribute value of branched; and

a plurality of attributes for the branch entity, the attributes comprising

a series identifier having a value representing a meeting identifier of the first one of the videoconference meeting entities,
a branch identifier having a value representing a meeting identifier of the second one of the videoconference meeting entities, and

an index of a videoconference meeting represented by the second one of the videoconference meeting entities.

3. A method for displaying a schedule of the videoconference meetings, which includes the data structure of claim 2, the method comprising:

selecting a coverage period for the schedule of meetings, the coverage period comprising a plurality of intervals; and

for each of the intervals in the coverage period, and each of the videoconference meeting entities having an interval type attribute value of repeating, displaying an indicator of the occurrence of the respective videoconference meeting when the occurrence of the respective videoconference meeting intersects temporally with the interval, and none of the series identifiers of the branch entities has a value representing the meeting identifier of the videoconference meeting entity representing the respective videoconference meeting.

4. The method of claim 3, further comprising:

determining whether the occurrence of the respective videoconference meeting intersects temporally with the interval, comprising at least one step selected from the group comprising
determining whether the scheduled start time of the respective videoconference meeting follows an end time of the interval, and
determining whether the scheduled end time of the respective videoconference meeting precedes a start time of the interval.

5. A medium or waveform containing a program of instructions adapted to perform the method of claim 3.

6. An apparatus to perform the method of claim 3.

7. The data structure of claim 2, further comprising:
a plurality of media entities each representing a link between a media file and one of the videoconference meeting entities.

8. A method for managing media entities, which includes the data structure of claim 2, the media entities each representing a link between a media file and one of the videoconference meeting entities, the method comprising:
storing one of the media files;
generating a first one of the media entities representing a link between the one of the media files and a selected one of the videoconference meeting entities; and
generating one or more second ones of the media entities each representing a link between the one of the media files and one of the branch entities having series identifier values representing the meeting identifier of the selected one of the videoconference meeting entities.

9. The method of claim 8, further comprising:
deleting the one of the media files when the first one of the media entities is deleted.

10. The method of claim 8, further comprising:

retaining the one of the media files when any of the second ones of the media entities are deleted.

11. A medium or waveform containing a program of instructions adapted to perform the method of claim 8.


13. The data structure of claim 1, further comprising:
a plurality of invited videoconference participant entities each representing a videoconference participant invited to one of the videoconference meetings.

14. A method for calculating a videoconference participant load for an interval in a videoconference schedule, which includes the data structure of claim 13, the method comprising:
generating an array comprising, for each of the videoconference meetings having an occurrence that temporally intersects the interval,
generating a first data pair comprising a first participant value representing the number of the participant entities associated with the videoconference meeting.
and a first difference between a reference time and the start time of the videoconference meeting, and
generating a second data pair comprising a second participant value representing the negative of the number of the participant entities associated with the videoconference meeting and a second difference between the reference time and the end time of the videoconference meeting; and
traversing the array in order according to the first and second participant values to a participant total.

15. The method of claim 14, further comprising:
indicating a videoconference load violation when the videoconference participant total exceeds a predetermined maximum participant total.

16. A method for calculating a videoconference participant load for an interval in a videoconference schedule comprising a plurality of the videoconference meetings, the method comprising:

generating an array comprising, for each of the videoconference meetings having an occurrence that temporally intersects the interval,
generating a first data pair comprising a first participant value representing the number of videoconference participants associated with the videoconference meeting and a first difference between a reference time and the start time of the videoconference meeting, and

generating a second data pair comprising a second participant value representing the negative of the number of the videoconference participants associated with the videoconference meeting and a second difference between the reference time and the end time of the videoconference meeting; and
traversing the array in order according to the first and second differences, comprising adding the first and second participant values to a participant total.

17. The method of claim 16, further comprising:
indicating a videoconference load violation when the videoconference participant total exceeds a predetermined maximum participant total.

18. A medium or waveform containing a program of instructions adapted to perform the method of claim 16.

19. An apparatus to perform the method of claim 16.

* * * * *